

<https://helda.helsinki.fi>

---

## Enhancing the Web With Advanced Engineering

Carlos Preciado, Juan

2018-11

---

Carlos Preciado , J , Hernandez , J , Mikkonen , T & Klamma , R 2018 , ' Enhancing the Web With Advanced Engineering ' , Journal of Web Engineering , vol. 17 , no. 6-7 , pp. 399-404 . <https://doi.org/10.13052/jwe1540-9589.17671>

---

<http://hdl.handle.net/10138/310375>

<https://doi.org/10.13052/jwe1540-9589.17671>

---

unspecified

publishedVersion

---

*Downloaded from Helda, University of Helsinki institutional repository.*

*This is an electronic reprint of the original article.*

*This reprint may differ from the original in pagination and typographic detail.*

*Please cite the original version.*

---

# Enhancing the Web With Advanced Engineering

---

Juan Carlos Preciado<sup>1</sup>, Juan Hernández<sup>1</sup>, Tommi Mikkonen<sup>2</sup> and  
Ralf Klamma<sup>3</sup>

<sup>1</sup>*QUERCUS Software Engineering Group School of Technology,  
University of Extremadura, Spain*

<sup>2</sup>*University of Helsinki, Finland*

<sup>3</sup>*RWTH Aachen University, Germany*

*E-mail: jcpreciado@unex.es, juanher@unex.es,  
tommi.mikkonen@helsinki.fi, klamma@dbis.rwth-aachen.de*

Received October 2018;  
Accepted February 2019

In the last fifteen years, services deployed via the Web have become a part of our everyday lives. Supporting Web technologies have also evolved very quickly to respond to the new services and features demanded by mobile and always connected users. Currently, the demand for services on the Web is still growing at a very fast pace as well as new techniques, methods and cutting-edge technologies that make the Web an increasingly complex environment.

In this scenario, the application of systematic, disciplined and quantifiable approaches to development, operation, and maintenance of Web-based applications continues to have a full validity like never before. So, at this point, is where the Web Engineering [1] discipline comes into the scene. Web Engineering deals with the process of developing, deploying and maintaining Web applications. The main issues of Web Engineering encompass how to successfully manage the

*Journal of Web Engineering, Vol. 17\_6&7, 399–404.*

doi: 10.13052/jwe1540-9589.17671

© 2019 River Publishers

diversity and complexity of Web development, and to avoid potential failures that may have serious implications [2].

During the last decades, the Web Engineering research community has handled innovative concepts such Model-Driven Web Engineering [3], Rich Internet Applications [4], Social Applications [5], Mobile Web Apps [6] or Quality Aspects [7], among other relevant goals. However, new challenges appear in Web engineering for now and the next few years with the incorporation of new trends that will be reflected in the Web as Artificial Intelligence [8], Machine Learning [9], Web Big Data and Web Data Analytics [10], Web of Things applications [11], new Computing and Deployment approaches [12], among other significant challenges.

As one of the references in this research field, the International Conference on Web Engineering (ICWE), is the prime yearly international conference on the different aspects of designing, building, maintaining and using Web applications. With the ever-increasing importance of the Web in almost all walks of life, ICWE bring together practitioners and researchers to tackle the emerging challenges in the engineering of Web applications.

This special issue in the Journal of Web Engineering is oriented for discussing on advanced techniques, new challenges, current state-of art and experiences for Enhancing the Web with Advanced Engineering dealing with a selection of the best papers of the 2018 edition of ICWE [13].

Gallidabino *et al.* extend the standard HTML5 WebWorker API with the concept of liquid WebWorkers (LWWs), so that developers can transparently offload parallel execution of stateless tasks by managing the necessary device selection and direct peer-to-peer data transfer. By introducing the liquid WebWorker API into our Liquid.js framework, they present how to create a pool of devices sharing their CPU processing capabilities according to different policies. They propose the design of liquid WebWorkers and their implementation within the Liquid.js framework. LWWs are designed for building liquid application featuring heavy computations, which are dynamically redeployed across multiple, partially idle heterogeneous devices. The goal is to avoid slowing down the overall performance of the application because

some of its components are running on a slow, bottleneck device. The preliminary evaluation of the liquid WebWorkers concept and their prototype implementation shows that there is the opportunity for increasing the overall performance of liquid Web applications when LWWs are migrated from slow to more powerful devices. Overall, with the extension of Liquid.js presented in their paper, they can create complex device ensembles able to directly connect and transparently share storage and computation resources between them. This allows developers to control how their liquid Web applications balance performance and energy consumption.

On the other hand, Govind *et al.* present a novel concept called “Semantic Finger-printing” that allows Web content classification solely based on the information derived from the named entities contained in a Web document. It encodes the semantic nature of Web content into a concise vector, namely the semantic fingerprint. Thus, they expect that semantic fingerprints, when utilized in combination with machine learning, will enable a fine-grained classification of Web contents. In order to empirically validate the effectiveness of semantic fingerprinting, they also perform a case study on the classification of Wikipedia documents. Even further, they thoroughly examine the results obtained by analyzing the performance of Semantic Finger-printing with respect to the characteristics of the data set used for the experiments. In addition, they also investigate performance aspects of the engineered approach by discussing the run-time in comparison with its competitor baselines. They observe that the semantic fingerprinting approach outperforms the state-of-the-art baselines as it raises Web contents to the entity-level and captures their core essence. Moreover, this approach achieves a superior run-time performance on the test data in comparison to competitors.

Additionally, Bakaev *et al.* describe a visual analysis tool which takes a WUI screenshot and produces structured and machine-readable representation (JSON) of the interface elements’ spatial allocation. The implementation is based on OpenCV (image recognition functions), dlib (trained detector for the elements’ classification), and Tesseract (label and content text recognition). The JSON representation is used to automatically calculate several metrics related to visual complexity,

which is known to have major effect on user experience with UIs. They further describe a WUI measurement platform that allows integration of the currently dispersed sets of metrics from different providers and demonstrate the platform's use with several remote services. They perform statistical analysis of the collected metrics in relation to complexity-related subjective evaluations obtained from 63 human subjects of various nationalities. Finally, they build predictive models for visual complexity and show that their accuracy can be improved by integrating the metrics from different sets. Regressions with the single index of visual complexity metric that they proposed had  $R^2 = 0.460$ , while the best joint model with 4 metrics had  $R^2 = 0.647$ .

Complementary, Toala *et al.* present a conceptual model (CM) for the study of the effect of personality and productivity on the perceptions and adoption intentions regarding MDWE. Also, it has presented an exploratory observational study to assess whether there are significant differences between groups of subjects scoring at the high vs low level of the personality and productivity IVs with respect to the perception and intention DVs. This purpose with these two contributions is fostering the discussion among the research community regarding individual differences and how they may be affecting the use of MDWE methods and tools. They propose that, in order for the MDWE research community to come up with a CM that truly reflects the MDWE method adoption influencers, substantial empirical research needs to be done. In this sense its paper is only another drop in the bucket. Only with much more extensive data set will it be possible to apply much more sophisticated statistical analysis that serve to validate alternative CMs in order to better understand which are the real influencers of MDWE adoption.

Finally, Koren *et al.* conceptualize different methods to move video delivery from centralized cloud infrastructures to end user devices. They discuss their strengths and weaknesses and present design considerations. To exemplify a particular approach, the authors showcase the implementation and evaluation of their system called OakStreaming. It streams videos peer-to-peer via WebTorrent in HTML5. Particularly, they offer Web video providers a library that has various parameters, for instance to limit the bandwidth available for peer-to-peer uploads.

## References

- [1] Murugesan, S. and Deshpande, Y. *Web Engineering*, Lecture Notes in Computer Science Hot Topics, vol 2016, Springer Verlag, 2001
- [2] Y. Deshpande, S. Murugesan, A. Ginige, S. Hansen, D. Schwabe, M. Gaedke, and B. White. *Web Engineering* vol 1 2002.
- [3] Martínez, Y., Cachero, C. and Meliá, S. MDD vs. traditional software development: A practitioner's subjective perspective. *Information & Software Technology* 2013, 55, 189–200. doi:10.1016/j.infsof.2012.07.004.
- [4] Toffetti, G., Comai, S., Preciado, J., and Trigueros, M. L. (2011). *State-of-the art and trends in the systematic development of rich internet applications*. *Journal of Web Engineering (JWE)* 10, 1, 70–86.
- [5] Gavin, B. *Building social web applications*. Farnham: O'Reilly. 2009.
- [6] Charland, Andre and Brian Leroux (2011). *Mobile Application Development: Web vs. Native*. In: *Communications of the ACM* 54(5), 49–53. DOI: 10.1145/1941487.1941504.
- [7] Li, Y. N., Tan, K. C. and Xie, M. (2002). *Measuring web-based service quality*. In: *Total Quality Management* 13(5), S. 685–700. DOI: 10.1080/0954412022000002072.
- [8] Huimin Lu, Yujie Li, Min Chen, Hyoungseop Kim, and Seiichi Serikawa. *Brain Intelligence: Go beyond Artificial Intelligence*. *Mob. Netw.* 2018. DOI: <https://doi.org/10.1007/s11036-017-0932-8>.
- [9] Fernández-García, A.J., Iribarne, L., Corral, A., Criado, J. and Wang, J. *A recommender system for component-based applications using machine learning techniques*, *Journal of Knowledge-Based Systems*, Volume 164, 2019, <https://doi.org/10.1016/j.knosys.2018.10.019>.
- [10] Alonso-Betanzos, A., Gámez, J.A., Herrera, F., Puerta, J.M. and Riquelme, J. *Volume, variety and velocity in Data Science*, *Journal of Knowledge-Based Systems*, Volume 117, 2017, <https://doi.org/10.1016/j.knosys.2016.11.005>.

- [11] Vandewalle, Jean-Jacques (2009). *The Web of Things: Interconnecting Devices with High Usability and Performance*. In: Proceedings of the International Conference on Embedded Software and Systems, S. 323–330.
- [12] Benlian, A., Kettinger, W., and Sunyaev, A., *The Transformative Value of Cloud Computing: A Decoupling, Platformization, and Recombination*. Journal of Management Information Systems. 2018. <https://doi.org/10.1080/07421222.2018.1481634>
- [13] International Conference on Web Engineering. 2018. <https://icwe2018.webengineering.org/>